conf. 790113--4

LA-UR -78-1915

MASTER

TITLE: A NETWORK FOR DATA ACQUISITION IN AN EXPERIMENTAL LABORATORY

AUTHOR(S): CHARLES P. MILICH, AP-4
and
ANDREW J. KOZUBAL, E-5

SUBMITTED TO: Twelfth Hawaii International Conference on System Sciences at the University of Hawaii on January 4 and 5, 1979.

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87545

An Affirmative Action/Equal Opportunity Employer

Form No. 836 R2
St. No. 2629
1/78

DEPARTMENT OF ENERGY
CONTRACT W-7405-ENG. 36

A NETWORK FOR DATA ACQUISITION IN AN EXPERIMENTAL LABORATORY

by

Charles P. Milich, Staff Member, AP-4, MS-567

and

Andrew J. Kozubal, Staff Member, E-5, MS-582

Los Alamos Scientific Laboratory
University of California
Los Alamos, New Mexico 87545
Phone: (505)-667-7100
FTS - 8-843-7100

ABSTRACT

The need for intercomputer communications is expanding, particularly in the area of the experimental laboratory. Satellite computers used for data preprocessing or plant control must be capable of communications with a host computer. This paper describes one such solution to provide the linkage between a single host computer and a multitude of satellite computers. The use of fiber optics and modem linkages are covered and the advantages of fiber optics in a high EMI environment are demonstrated.

I. INTRODUCTION

The Applied Photochemistry Division at the Los Alamos Scientific Laboratory is conducting numerous experiments in laser spectroscopy, laser induced fluorescence, and photochemical reactions. Many of these experiments use computer-based data acquisition systems to collect and display data. Because these systems have limited peripherals, most program development and data analysis is performed on a host processor. Recently, the data acquisition systems were linked to the host, thus forming a host-satellite network.

Ten satellite processors are in use as data acquisition systems. They consist of two NOVA 2, six NOVA 3, and two MicroNOVA processors, and these are linked to the host by various means. Two are linked by a high-speed data channel interface. This interface allows transfer rates of about one million bytes per second, but is limited to a range of 25 meters from the host.

Seven of the satellites are linked to the host via fiber optic cables. These links are full duplex and normally are used asynchronously at 9600 bits per second. The longest run for a fiber optic cable is 400 meters. One satellite processor uses a 1200 baud modem and dial-up telephone access to the host.

A minimum satellite consists of a processor with up to 64k words of main memory, an interface to the link, a terminal, and a CAMAC subsystem. A PROM-based network communications program is included with most satellites. Other peripheral equipment attached to some satellites are a dual diskette drive, and an electrostatic printer/plotter. Prior to installation of the communications links, diskettes were used to transfer program and data files between the satellites and the host.

The host processor is a Data General ECLIPSE S/230 with 448k bytes of error correcting main memory and a floating point processor. Peripherals include two ninety M byte disk drives, a dual diskette drive, a nine-track magnetic tape drive, and a sixteen-line programmable communications interface. The host processor runs Data General's Advanced Operating System (AOS), a multi-user, multiprocess system with extensive file and system protections. In this installation, as many as sixteen users may log onto and use AOS simultaneously through the communications interface. When performing data acquisition, the satellites use Data General's Real Time Operating System (RTOS) or Diskette Operating System (DOS). The AOS FORTRAN-IV compiler produces code which is compatible with RTOS and DOS. However, different run-time libraries must be used for programs targeted for each operating system.

II. NETWORK OPERATION

A user at a satellite gains access to AOS by entering a small program into main memory using a simple hardware bootstrap procedure. This program - AOSCOM - effectively connects the user terminal to the link, and permits the user to log on. AOSCOM also permits down-line loading of a user program to the satellite.

The AOS file system is remotely accessible from the user program at the satellite. A remote file access protocol permits creation, deletion, and read/write access to one or more files. Another feature of the protocol allows a user program at the satellite a full duplex communication path to a user co-process on the host. Hence, the satellite might collect and prefilter data, send the data to an AOS file, and instruct the co-process to perform further data analysis or filtering. The co-process would then put it's results in a data file and notify the satellite that it has completed its task. The satellite program would then read the new data file and display the results locally. These operations could proceed asynchronously so that data acquisition would not necessarily be interrupted.

III. NETWORK SOFTWARE

An AOS program, DWNLD, performs the down-line load and remote file access at the host. The user initiates this action by logging onto AOS and typing "DWNLD". DWNLD requests the user to enter the name of the program to be down-line loaded. When the down-line load operation is completed, DWNLD is ready to accept remote file access requests from the satellite.

The remote file access software was implemented at the FORTRAN level. The naming and call conventions were made similar to these used by Data General FORTRAN-IV. This was accomplished by taking the standard FORTRAN subroutine names and prefixing them with the letter "U" for the equivalent remote file access routines. For instance, the routine to open a file is named "OPEN"; hence, the routine to open a remote file is named "UOPEN".

When a remote file routine is called a request block is sent to DWNLD at the host. The request block is interpreted and an acknowledge or error block is sent back to the satellite. If the request is a read or write, the data block is transferred next. Data blocks may be sent in binary or ASCII modes, and files may be accessed sequentially or randomly.

To permit a user to test portions of his program on the host, a set of "U" routines were implemented which are equivalent to the standard FORTRAN routines included with AOS.

The remote file access software was implemented such that no changes to the operating systems were required and permitting compatibility with future revisions or releases of the operating systems. However, this does place some restrictions on the versatility of the software. First, the user may not use remote file access through the standard in-line FORTRAN statements such as "READ (unit, format)". Second, other languages such as BASIC cannot use this software. These restrictions are acceptable to nearly all users.

IV. LINK CONTROL SOFTWARE

A second, lower level protocol is being developed to perform handshake and error control on the links. By design, this level will have little or no affect on the current remote file access software. The link control protocol uses character transparency to distinguish control characters from binary data. A transmission is preceded by the character sequence DLE/STX, and terminated by DLE/ETX and a two-byte CRC. A transmission is acknowledged by DLE/ACK (or DLE/NAK if a CRC error is detected). Since the assumed mode of transmission is binary, no lateral parity check is performed.

V. AOSCOM IMPLEMENTATION

AOSCOM has four operating modes; (1) initialization, (2) communications, (3) program load, and (4) octal debug. The initialization mode is entered first. It's functions are to size RAM memory, perform a quick hardware diagnostic checkout, and enter the communications mode. The communications mode connects the user terminal to the communications link, allowing the user to log onto AOS. Character buffering is performed in both directions so that the terminal and link may operate at different speeds. A special control character sequence typed on the terminal puts AOSCOM into the octal debug mode. This mode permits insertion of breakpoints, examination and replacement of RAM and register contents, and execution of CAMAC commands.

The communications mode can also be interrupted by a two-character control sequence from the link. This sequence is sent from the host by DWNLD, and puts AOSCOM in the program load mode. AOSCOM does not perform the full program load, but instead loads a small communications loader into the highest 512 bytes of RAM.

On the NOVA 2 and 3 processors, AOSCOM resides in PROM on the I/O bus, and is copied into RAM by the hardware bootstrap loader (initiated by a front panel switch). It relocates itself to the top 2048 bytes of RAM and enters the initialization mode. The MicroNOVA version of AOSCOM also resides in PROM, but occupies the highest possible 2048 bytes of main memory. On power-up, the initialization mode is automatically entered. Also, whenever a HALT instruction is executed, the hardware in the MicroNOVA forced the debug mode to entered.

VI. CONCLUSIONS

Although many features of this network - hardware and software - are still being refined, some preliminary observations have been made. First, most hardware problems were connected with the diskette drives and media. With the communications links implemented, the diskette drives are no longer required except in a few experiments where the link is too slow to handle the high data rates. Second, when a satellite is not performing data acquisition, it's terminal may be used for program development or data reduction under AOS. This reduces the number of terminals required, since there are always one or two satellites not in use. Third, the links simplify the program generation and maintenance tasks of the system support team. The latest version of all general-purpose data acquisition programs are kept in a utility directory. Each of these files has it's access controls set to allow read and execute access by all users. Before the links were implemented, there was no way to guarantee that a user had the latest version of a program on his diskette. Also, diskettes given to a user had a habit of becoming lost or damaged. Fourth, the procedures for writing and testing of programs has been simplified, since the updated program file does not have to be copied to diskette whenever a change is made.

One final observation can be made. There is a high level of commonality - hardware and software - throughout the system, simplifying the hardware and software maintenance tasks. Hence, a satellite may be reconfigured and new application software written in a very short time. This is a common occurrence, since the experiments have constantly changing requirements.